REMARKS

Claims 1, 2, 4, 5, 8, 9, 11, 12, 15, 21, and 23 are amended to more clearly point out that which is being claimed.

Claims 6-7, 13-14, 17-20, 24 are cancelled without prejudice.

Claims 25-33 have been added to further point out that which is being claimed.

Claims 1-5, 8-12, 15, 16, 21-23, and 25-28 are pending.

In the specification, the "Related Applications" paragraph on page 1, beginning on line 2, has been amended to include the serial number, filing date and title of a related patent application having attorney docket number of MS1-710US.

Rejections under 35 U.S.C. §103(a)

Of the pending claims, **Claims 1-4, 8-10, 15, 16, and 21-23** stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,649,196 issued to Woodhill et al. (*Woodhill*) and further in view of U.S. Patent No. 6,314,565 issued to Kenner et al. (*Kenner*). Applicants respectfully traverse these rejections for at least the following reasons.

Woodhill is directed towards distributed processing backup file systems and discloses that individual binary objects that are stored and backed-up periodically

can be assigned unique identifiers. *Kenner* discloses that script files can be used in a comparative way to determine which multimedia files to download.

Claim 1 is directed towards a software version control method that includes assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups, and for each downloadable file group compressing together all assigned data files to form one processed image for the downloadable file group. The method also includes associating each resulting processed image with a unique identifier, generating a listing of unique identifiers, and storing the processed images and the listing of unique identifiers within a source device. The method also includes comparing the listing of unique identifiers with a current listing of unique identifiers in a client device, and selectively downloading processed images from the source device to the client device whose unique identifiers appear in the listing of unique identifiers but not in the current listing of unique identifiers in the client device.

Neither *Woodhill* nor *Kenner*, alone or together, disclose or suggest the software version control method in Claim 1. For example, the method includes assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups and that for each downloadable file group the method includes compressing together all assigned data files to form one processed image for the downloadable file group. *Woodhill* does not do this, but instead gives each binary object its own identifier. *Kenner* does not do this but rather deals only with specific individual software components.

The method also includes, for example, associating each resulting processed image with a unique identifier, generating a listing of unique identifiers, storing the processed images and the listing of unique identifiers within a source device and comparing the listing of unique identifiers with a current listing of unique identifiers in a client device so as to selectively download processed images from the source device to the client device whose unique identifiers appear in the listing of unique identifiers but not in the current listing of unique identifiers in the client device. Since neither *Woodhill* and/or *Kenner* deal with such processed images, they also fail to disclose the ability for the software version control method to conduct a list comparison of unique identifiers for such processed images (i.e., compressed groups of related data files).

In **Claim 2**, which depends from Claim 1, the source device is further specified to include at least one server device.

Claim 3, which depends from Claim 1, further recites that each unique identifier is derived from its corresponding processed image.

Claim 4, which depends from Claim 1, further specifies that assigning data files to downloadable file groups includes assigning a plurality of related function data files to one downloadable file group.

Claim 25, which depends from Clam 1, further recites that the one processed image for the downloadable file group has a ".cim" extension.

Independent **Claim 8** is directed towards a computer-readable medium having computer-executable instructions for causing at least one processing unit to

perform certain acts. The acts include assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups and for each downloadable file group compressing together all assigned data files to form one processed image for the downloadable file group. Further acts recited include associating each resulting processed image with a unique identifier, generating a listing of unique identifiers, and storing the processed images and the listing of unique identifiers within a source device. Additional acts include comparing the listing of unique identifiers with a current listing of unique identifiers in a client device, and selectively downloading processed images from the source device to the client device whose unique identifiers appear in the listing of unique identifiers but not in the current listing of unique identifiers in the client device.

The exemplary arguments stated above with regard to Claim 1 are also applied to Claim 8.

Claim 9, which depends from Claim 8, further recites that the source device includes at least one server device.

In **Claim 10**, which depends from Claim 8, it is further specified that each unique identifier is derived from its corresponding processed image.

Claim 26, which depends from Clam 8, further recites that the one processed image for the downloadable file group has a ".cim" extension.

Independent **Claim 15** is directed towards an apparatus that includes memory and logic. The logic is operatively configured to assign each of a plurality of data files to one of a plurality of specific corresponding downloadable

file groups, and for each downloadable file group compress together all assigned data files to form one processed image for the downloadable file group. The logic also associates each resulting processed image with a unique identifier and stores the processed images and a listing of unique identifiers to the memory. The logic is also configured to compare the listing of unique identifiers with a current listing of unique identifiers in a client device to identify processed images that need to be provided to the client device.

The exemplary arguments stated above with regard to Claim 1 are also applied to Claim 15.

Claim 16, which depends from Claim 15, further recites that each unique identifier is derived from its corresponding processed image.

Claim 27, which depends from Clam 15, further recites that the one processed image for the downloadable file group has a ".cim" extension.

Independent **Claim 21** is directed towards a system that includes a network, a server device and a client device. The server device is configured to assign each of a plurality of server-based data files to one of a plurality of specific corresponding server-based downloadable file groups, and for each server-based downloadable file group compress together all assigned data files to form one processed image for the server-based downloadable file group. The server device is also configured to associate each resulting processed image with a unique identifier. The server device can also selectively output the processed images and a latest listing of unique identifiers over the network. The client device, which

communicates with the server device through the network, is configured to maintain a listing of unique identifiers associated with processed images stored locally within the client device and to compare the listing of unique identifiers with a downloaded latest listing of unique identifiers from the server device, and selectively download processed images whose unique identifiers appears in the latest listing of unique identifiers from the server device but not in the listing of unique identifiers in client device.

The exemplary arguments stated above with regard to Claim 1 are also applied to Claim 21.

Claim 22, which depends from Claim 21, further specifies that each unique identifier is derived from its corresponding processed image.

In **Claim 23**, which also depends from Claim 21, the server device recited as being further configured to selectively assign a plurality of related function data files to one downloadable file group.

Claim 28, which depends from Clam 25, recites that the one processed image for the server-based downloadable file group has a ".cim" extension.

Of the pending claims, **Claims 5 and 12** stand rejected under 35 U.S.C. §103(a) as being unpatentable over *Woodhill* in view of *Kenner* and further in view of U.S. Patent No. 5,848,565 issued to Cowan (*Cowan*). Applicants respectfully traverse these rejections for at least the following reasons.

Cowan is added to *Woodhill* and/or *Kenner* to add persistent memory to the mix.

Cowan does not, however, add the missing elements/features as recited in independent Claims 1 and 8, from which Claims 5 and 12, respectively, depend.

More specifically, **Claim 5** is directed towards a software version control method that includes assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups, and for each downloadable file group compressing together all assigned data files to form one processed image for the downloadable file group. The method also includes associating each resulting processed image with a unique identifier, generating a listing of unique identifiers, and storing the processed images and the listing of unique identifiers within a source device. The method also includes comparing the listing of unique identifiers with a current listing of unique identifiers in a client device, and selectively downloading processed images from the source device to the client device whose unique identifiers appear in the listing of unique identifiers but not in the current listing of unique identifiers in the client device. The method further includes sending the processed image and the listing of unique identifiers to a client device that stores the processed image and the listing of unique identifiers in a persistent memory

Neither *Woodhill, Kenner,* or *Cowan,* alone or combined, disclose or suggest the software version control method in Claim 5. The arguments presented above with respect to Claim 1 apply here too, for example.

Claim 12 is directed towards a computer-readable medium having computer-executable instructions for causing at least one processing unit to

perform certain acts. The acts include assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups and for each downloadable file group compressing together all assigned data files to form one processed image for the downloadable file group. Further acts recited include associating each resulting processed image with a unique identifier, generating a listing of unique identifiers, and storing the processed images and the listing of unique identifiers within a source device. Additional acts include comparing the listing of unique identifiers with a current listing of unique identifiers in a client device, and selectively downloading processed images from the source device to the client device whose unique identifiers appear in the listing of unique identifiers but not in the current listing of unique identifiers in the client device. Additional acts include sending the processed image and the listing of unique identifiers to a client device that stores the processed image and the listing of unique identifiers in a persistent memory.

Neither *Woodhill*, *Kenner*, or *Cowan*, alone or combined, disclose or suggest the computer-readable medium as recited in Claim 5. The arguments presented above with respect to Claims 1 and 8 apply here too, for example.

Independent **Claim 29** has been added, along with dependent **claims 30-33.** Claim 29 is directed towards a computer-readable medium having computer-executable instructions for causing at least one processing unit to perform acts that include assigning each of a plurality of data files to one of a plurality of specific corresponding downloadable file groups, for each downloadable file group, compressing together all assigned data files to form one processed image for the downloadable file group, associating each resulting processed image with a unique

identifier, generating a listing of unique identifiers, and storing the processed images and the listing of unique identifiers within a source device.
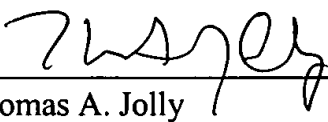
The cited art neither discloses nor reasonably suggests this novel computer-readable medium, or those in Claims 30-33. The arguments presented above with respect to Claims 1 and 8, for example, apply here too.

Conclusion

For at least the reasons presented above, the pending claims are clearly patentable over the cited art. It is respectfully requested, therefore, that the rejections be reconsidered and withdrawn and the application be allowed.

Respectfully Submitted,

Dated: 6/12/2003   By: _____
                              Thomas A. Jolly
                              Reg. No. 39,241
                              (509) 324-9256